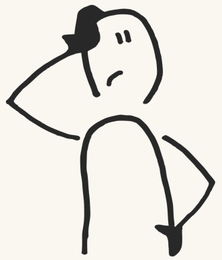
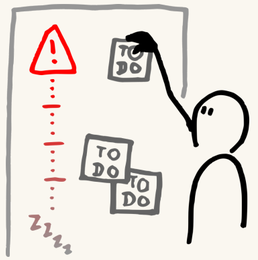
PYTHON BASIC

# Final Project Presentation

JUE JUE LA WON

# Project Overview






## project title(PureSkinMart store)

This Python project simulates a basic store and personal banking system with user interaction via the command line. It allows a user to log in, access a personal bank, shop for products, upgrade to premium, and even generate a reward code. The goal is to practice using Python conditionals, loops, functions, and basic logic to build a working mini-system.

## python concepts used:

- if-else conditions for logic branching
- while True loops for repeating menus
- break to exit loops
- input() and print() for user interaction
- try-except to safely handle wrong inputs (e.g., when entering numbers)
- lambda function for reward code generation
- lists and dictionaries to store valid passwords and product info
- lower() to make input checking more user-friendly
- Arithmetics operator to calculate the banking user's input

## LOGIN IN :THIS PART OF THE PROJECT HANDLES THE INITIAL LOGIN PROCESS FOR USERS BEFORE ACCESSING THE PURESKINMART SYSTEM.

input

practice > ...

```python
#practice
total_amount=0
print('Enter 1 for login to account ')
position=input('Enter :') #ask user to login (that is first reqriment that every user have to do)
position=position.lower()# covert input to lowercase
if position =='1':
        name=input('enter your name :')
        password=int(input(' (password must contain 6 numbers) \nenter your password  : '))


        password=[11111,22222,33333,44444,55555] #passward list


        if len(password) < 5:#check if passward is less than 6
            print('your password must be at least 6 number long.')
        else:
            print('your are accepted \n welcome to PureSkinMart store \n')

```

output

```
PS C:\Users\User\PycharmProjects\PythonProject\python> & "C
Enter 1 for login to account
Enter :1
Enter your name: moe
Enter your password (must be 6 digits): 1111
 Your password must be exactly 6 digits.
Enter your password (must be 6 digits): 111111
 You are accepted!
Welcome to PureSkinMart store.
Enter b for personal bank

Enter s for go to store

Enter p for go to my  personal's profile

Enter q for quit

Enter your perfence :
```

If the user chooses to log in, the system then asks for their name and a password. The password must be a (6-digit number). If the user enters fewer than 6 digits or a password that is not in the list of valid passwords, the program will display an error message and replay the "enter your password" again. This helps ensure that only authorized users can access the store.

# THE HOMEPAGE OF STORE & PERSONAL BANK : AFTER THE USER SUCCESSFULLY LOGS INTO THE SYSTEM, THEY ARE PRESENTED WITH A MAIN MENU THAT OFFERS FOUR OPTIONS

## INPUT

py | 7+8 | 11+12 | practice | extra | 13+14 | 15+16 | 9+10

actice > ...

The system keeps showing this menu until the user chooses to quit, allowing them to explore all the features. Or choose "q "will end

```
#main loop
while True:
    #show main menu
    print("Enter b for personal bank\n")
    print('Enter s for go to store\n')
    print('Enter p for go to my  personal\'s profile\n')
    print('Enter q for quit\n')
    position=input('Enter your perfence :')
    position=position.lower()

    if position=='b':
        user_name=input('Enter your name\n')
        valid_passwords = [111111, 222222, 333333, 444444, 555555]  # Valid passwords

        try:
            password = int(input('Enter your account password:\n'))  # Try converting input to integer
            if password in valid_passwords:  # Check if password is valid(try except)
                print('Welcome to PureSkinMart personal bank')
            else:
                print('Incorrect password. Please try again.')
        except ValueError:#catch non-numeric input
            print('Invalid input! Password must contain only numbers. \n your actual password from login account.')


        while True:#bank loop
                print('Enter d for deposit\n')
                print('Enter w for withdraw\n')
                print('Enter c for check total amount\n')
                print('Enter q for quit\n')
```

In here everything is the same as the login in but in password function I uses a try-except block to safely handle password input. When the user is asked to enter" their numeric" password, the program attempts to convert the input into an integer using int(). If the user enters something that is not a number (like letters or symbols), this would normally cause an error and crash the program. But by using try-except, the program catches that mistake and shows a helpful message like “Invalid input! Password must be numeric.” This helps prevent crashes and gives the user a chance to try again without restarting the whole program.

output with error

```
Enter your perfence :b
Enter your name
moe
Enter your account password:
1111rs
Invalid input! Password must contain only numbers.
 your actual password from login account.
```

*output*

```
Enter your perfence :b
Enter your name
moe
Enter your account password:
111111
Welcome to PureSkinMart personal bank
Enter d for deposit

Enter w for withdraw

Enter c for check total amount

Enter q for quit
```

# PERSONAL BANK INPUT

7+8 | 11+12 | practice | extra | 13+14 | 15+16 | 9+10 | work.py

```
while True:#bank loop
        print('Enter d for deposit\n')
        print('Enter w for withdraw\n')
        print('Enter c for check total amount\n')
        print('Enter q for quit\n')

        user_option=input('Enter your option :')
        if user_option=='d':
            deposit=int(input('Enter deposit amount:'))
            if deposit<=0:
                print('Invalid amount')
            else:
                total_amount += deposit
                print('Your deposit amount is ', deposit, '\nYour total amount is ', total_amount)
        elif user_option=='w':
                withdraw=int(input('Enter your withdraw amount :'))
                if withdraw<=0:
                    print('Invalid amount.')
                elif withdraw>total_amount:
                    print('Insufficient amount')
                elif withdraw>0 and withdraw<=total_amount:
                    total_amount-=withdraw
                    print('You withdrawed',withdraw,'your remaining amount is ',total_amount)
                else:
                    print('Invalid Option')

        elif user_option=='c':
                print('User name :',user_name,'\nUser passwordb :',password,'\nTotal amount:',total_amount)
                print('if you woul love to set up to use your bank card trougth payment with our personal bank\n please content with bank
        elif user_option=='q':
                print('Thanks for choosing our Bank')
                break
```

This line ensures that the user cannot withdraw more money than they currently have in their bank account. If the amount the user enters is greater than their total balance, the program shows a message like “Insufficient amount”. This prevents users from going into a negative balance and helps the system behave like a real-world banking application.(Also total amount is out of the while loop

When a user chooses to deposit or withdraw money, the program performs **real-time arithmetic** calculations using the user's input and also store the total amount of user for later .

## OUTPUT

```
Enter your option :c
User name : moe
User passwordb : 111111
Total amount: 5000
if you woul love to set up to use your bank card trougth payment with
 please content with bank for more information
Enter d for deposit

Enter w for withdraw

Enter c for check total amount

Enter q for quit

Enter your option :w
Enter your withdraw amount :50000
Insufficient amount
```

*This part of the program runs after a user has successfully logged into the personal bank section. It displays a menu of banking options inside a while True loop, allowing the user to choose what they want to do with their personal bank account offer by PureSkinMart store (just like a gift card you seen from most of the store you can buy with this bank account and card from PureSkinMart ) The options include depositing money, withdrawing money, checking their total amount, or quitting the banking system.(Just like kpay)*

# STORE :THIS SECTION ALLOWS THE USER TO EXPLORE THE PURESKINMART PRODUCT.

## INPUT

```
elif position=='s':
    while True:
        print('enter p for prouct list')# go for as a list in both proucts

        user_preference=input('enter p for to see the avaliable and restock product list:')
        if user_preference=='p':
            product_list=['innisfree','\nlaneige','\nsome by mi','\nnature repunic','\nmissha','\ncorsrx']
            print('available products:')
            for product in product_list:
                print(product)
        else:user_preference=='r'
        print('restock product list are:')
        restock_list= [ #list
            {"name": 'Benton', "category": 'skin care'},
            {"name": 'lancome' , "category" : 'lip care'},
            {"name" : 'dior' , "category" : 'lip oil'},
            {"name" : 'skin 1004' , "category" : 'serum & sun cream'}

        ]
        for product1 in restock_list:
            print(f"- {product1['name'].title()} ({product1['category']})")
        break
```

The restock_list is a list of dictionaries. Each dictionary contains:
Name and categories of each product

## OUTPUT

```
Enter your perfence :s
enter p for prouct list
enter p for to see the avaliable and restock product list:p
available products:
innisfree

laneige

some by mi

nature repunic

missha

corsrx
restock product list are:
- Benton (skin care)
- Lancome (lip care)
- Dior (lip oil)
- Skin 1004 (serum & sun cream)
thank you for shoping with us
PS C:\Users\User\PycharmProjects\PythonProject\python>
```

The store product part gives the user the ability to see available product and restock product

# UPGRADE TO PREMIUM & AVAILABLE BANKING SYSTEM FOR PAYMENT

7+8 | 11+12 | practice | extra | 13+14 | 15+16 | 9+10 | work.py

```
elif position=='p':
    while True:#premium loop
        print('enter u for upgrade to premium\n')
        print('enter c for check account\n')
        print('enter r for reward code\n')
        print('enter q for quit\n')
        decision=input('enter your preference:\n')
        if decision=='u':# bank payment for upgrade premium
            print("(Note \n Upgrade to Premium and enjoy 5% off every purchase, unlock exclusive discounts, \n earn a personal reward CODE to
            preference=input('Enter yes or no :')
            if preference=='yes':
                    print(" Welcome to the Premium Upgrade Portal")

                    user_name = input("Enter your full name: ")
                    email = input("Enter your email address: ")

                    print("\nChoose a payment method:")
                    print("1.personal Bank Transfer")
                    print("2. Credit/Debit Card")
                    print("3. Mobile Payment (e.g., Wave, KBZPay)")
                    method = input("Enter the number of your payment method: ")

                    if method == "1":
                        bank_number = input("Enter your bank account number: ")
                        bank_name = input("Enter your bank name: ")
                        print(f"\nThank you, {user_name}. We'll process your upgrade via {bank_name} account {bank_number}.")

                    elif method == "2":
                        card_number = input("Enter your card number (xxxx-xxxx-xxxx-xxxx): ")
                        expiry = input("Enter card expiry date (MM/YY): ")
                        cvv = input("Enter CVV (3 digits): ")
```

I will go over each method of payment in next slide .

```
                    elif method == "3":
                        mobile_id = input("Enter your mobile payment ID or phone number: ")
                        app_name = input("Enter the app name (e.g., WavePay, KBZPay): ")
                        print(f"\nThanks, {user_name}. We'll send a request via {app_name} to {mobile_id}.")

                    else:
                        print(" Invalid payment method selected.")

                    print("\n Premium upgrade request received. We'll verify your payment shortly!")

                else: decision=='no'
                print('thank you for taking time')
```

This part of the program allows users to upgrade to a premium membership in PureSkinMart. It guides users through the process of upgrading by collecting their name, contact information, and preferred payment method.
Once the user agrees to upgrade, the system confirms the action and gives a message that the upgrade request will be processed and if user choose 'no' it will go back to the asking enter your preference for upgrade,check,and reward code

# payment method

What It Does

## method(1)

```
u
(Note
 Upgrade to Premium and enjoy 5% off every purchase, unlock exclusive discounts,
 earn a personal reward CODE to share — get 2% every time someone uses your CODE!
Enter yes or no :yes
 Welcome to the Premium Upgrade Portal
Enter your full name: moe
Enter your email address: jjuju4453

Choose a payment method:
1.personal Bank Transfer
2. Credit/Debit Card
3. Mobile Payment (e.g., Wave, KBZPay)
Enter the number of your payment method: 1
Enter your bank account number: 123456780qwa
Enter your bank name: AyA

Thank you, moe. We'll process your upgrade via AyA account 123456780qwa.
```

- in here you can go with your personal bank offer by PureSkinMart or your chose of bank you like.

```
Enter yes or no :no
thank you for taking time
thank you for choosing us
```

- In here if you choose ' no ' to upgrade premium the program will end and go back to the personal profile loop

## Method(2)

```
Choose a payment method:
1.personal Bank Transfer
2. Credit/Debit Card
3. Mobile Payment (e.g., Wave, KBZPay)
Enter the number of your payment method: 2
Enter your card number (xxxx-xxxx-xxxx-xxxx): 112345rrfhe
Enter card expiry date (MM/YY): 4/2027
Enter CVV (3 digits): 123

Thank you, moe. Your card ending in rfhe will be used for the upgrade.
```

⊗ 0 ⚠ 0

- In method (2/3) I didn't limited the word or number in here to enter freely

## method(3)

```
Choose a payment method:
1.personal Bank Transfer
2. Credit/Debit Card
3. Mobile Payment (e.g., Wave, KBZPay)
Enter the number of your payment method: 3
Enter your mobile payment ID or phone number: 09963944070
Enter the app name (e.g., WavePay, KBZPay): kpay

Thanks, moe. We'll send a request via kpay to 09963944070.

 Premium upgrade request received. We'll verify your payment shortly!
```

⊗ 0 ⚠ 0

# CHECK THE ACCOUNT & REWARD CODE

```
        elif decision=='c':#check the account
            print('User name :',user_name,'\nUser passwordb :',password,'\nTotal amount:',total_amount)
        elif decision=='r':#reward code
            print('Enter your name to get a reward code:')
            user_name = input()
            import random#to generate number for reward code
            user_name = input("Enter your name to get a reward code: ")
            generate_code = lambda name: f"{name.upper()}{random.randint(1000, 9999)}"

            reward_code = generate_code(user_name)



            print("Your reward code is:", reward_code)
            break
        else:decision=='q'
        print('thank you for choosing us')
else:position=='q'
print('thank you for shoping with us')
break
```

In here I use random.randiant to generate random number for each reward code

```
c
User name : moe
User passwordb : 111111
Total amount: 0
thank you for choosing us
enter u for upgrade to premium

enter c for check account

enter r for reward code

enter q for quit

enter your preference:
r
Enter your name to get a reward code:
moe
Enter your name to get a reward code: moe
Your reward code is: MOE4319
thank you for shoping with us
PS C:\Users\User\PycharmProjects\PythonProject\python>
```

The Reward Code feature lets users generate a unique code based on their name. This code can be used in a store promotion system and check account feature use to check the user's information.( I added check account in both personal bank and here)

# QUIT

```
            import random#to generate number for reward code
            user_name = input("Enter your name to get a reward code: ")
            generate_code = lambda name: f"{name.upper()}{random.randint(1000, 9999)}"

            reward_code = generate_code(user_name)


            print("Your reward code is:", reward_code)
            break
        else:decision=='q'
        print('thank you for choosing us')
else:position=='q'
print('thank you for shoping with us')
break
```

Ln 43, Col 9  Spac

In my Python project, the "Quit" option lets the user exit the program. It makes sure the user can stop using the program when they want.

```
PS C:\Users\User\PycharmProjects\PythonProject\python> & "
Enter l for login to account
Enter :l
Enter your name: moe
Enter your password (must be 6 digits): 111111
 You are accepted!
Welcome to PureSkinMart store.
Enter b for personal bank

Enter s for go to store

Enter p for go to my  personal's profile

Enter q for quit

Enter your perfence :q
thank you for shoping with us
PS C:\Users\User\PycharmProjects\PythonProject\python>
```

# The End

Please feel free to share any suggestions

Apologies if the slide feels a bit too packed. I was trying to cover everything clearly